

TOWARDS PURE ADAPTIVE SEARCH

by

**W.P. Baritompa, R.H. Mladineo¹ G.R. Wood,
Z.B. Zabinsky¹ and Zhang Baoping**

University of Canterbury, Christchurch, New Zealand

No. 83

March, 1993

Abstract – The algorithm known as Pure Adaptive Search is a global optimisation ideal with desirable complexity. In this paper we temper it to a framework we term Somewhat Adaptive Search. This retains the desirable complexity, but allows scope for a practical realisation. We introduce a new algorithm termed Pure Localisation Search which attempts to reach the practical ideal. For a certain class of functions the gap is bridged.

Keywords – global optimisation, stochastic, random search, localisation, complexity

¹ Visitors to the Mathematics Department, University of Canterbury

Towards Pure Adaptive Search

W.P. Baritompa, R.H. Mladineo, G.R. Wood, Z.B. Zabinsky and Zhang Baoping

June 23, 1992

Abstract

The algorithm known as Pure Adaptive Search is a global optimisation ideal with desirable complexity. In this paper we temper it to a framework we term Somewhat Adaptive Search. This retains the desirable complexity, but allows scope for a practical realisation. We introduce a new algorithm termed Pure Localisation Search which attempts to reach the practical ideal. For a certain class of functions the gap is bridged.

Running head: Adaptive Search

AMS 1991 Subject Classification: 90C30, 65K05

Key words and phrases: global optimisation, stochastic, random search, localisation, complexity

1 Introduction

The ideal of Pure Adaptive Search (PAS) has been introduced and discussed in [6] and [13]. Pure Adaptive Search occurs when we are always able to choose the next evaluation point according to a uniform distribution on the improving region, or “level set”, of the feasible space. In [13] it was shown that when Pure Adaptive Search is applied to global mathematical programs satisfying the Lipschitz condition, the expected number of iterations to convergence increases at most linearly in the dimension of the problem, a desirable complexity result. Convergence here occurs when the lowest known value is within a given tolerance of the global minimum. A difficulty which immediately arises is that Pure Adaptive Search appears to be hard to realise in practice. Encouragement, however, comes from the observation that several other practical random search algorithms have reported linearity in dimension, for example [10], although only for convex programs.

Pure Adaptive Search can be implemented, albeit very inefficiently, by running Pure Random Search and accepting only those points which provide improved function evaluations. Two attempts have already been made to provide a more efficient implementation. These are the Improving Hit-and-Run algorithm [14], and the Hide-and-Seek algorithm [3]. The purpose of this paper is to approach the problem from a third perspective, which we now describe.

The central idea is to focus on an enlargement of the level set of Pure Adaptive Search. A delicate balancing act is required. The enlargement must be accessible in practice, yet small enough to retain the properties of PAS. Such an enlargement is provided by the deterministic algorithms for mathematical programs satisfying the Lipschitz condition found in [7, 9, 5, 12]. These have the property that, at each iteration, regions which cannot contain the global minima are stripped away from the domain. A general framework for such algorithms is described in [1]. Basso in [2] uses the word “localisation” for the resulting enlargement of the level set which is known to contain the global minimisers. Both [5] and [12] reduce to the well-known Piyavskii-Shubert algorithm, [7] and [9], for functions of a single variable. The localisations they provide reach towards the level set of PAS. Such algorithms, however, require an exponentially increasing number of function evaluations to reach convergence, as the dimension increases (see [8]). Convergence in [8] is measured using the distance between the lowest

evaluation so far and the lowest point of a lower envelope for the unknown function. This distance is called the “variation” in [12].

Is there an efficiently implementable algorithm, based on a stochastic variant of the Piyavskii-Shubert algorithm, which can realise the desirable complexity of PAS? This paper initiates a study of this question.

We begin by modifying PAS to an algorithm which we term Somewhat Adaptive Search (SAS). Somewhat Adaptive Search is a relaxation of Pure Adaptive Search, and is more likely to be efficiently implementable, yet still possesses the desirable complexity of PAS (Theorem 2.1 and Corollary 2.1). In the same breath, we modify the Piyavskii-Shubert style of algorithm to a stochastic search we term Pure Localisation Search (PLS). This algorithm chooses uniformly from the localisation. The convergence properties of PLS lie between PRS and PAS (Theorem 3.2 and Corollary 3.1). We conclude by showing that for a class of functions of a single variable, PLS does realise an SAS algorithm. This is shown in the last set of results (Theorem 4.1 and Corollary 4.1).

The layout of the paper is as follows. In §2 we introduce Somewhat Adaptive Search, prove the linear complexity result, and discuss the special case of “ ρ -adaptive” search. In §3 we define Pure Localisation Search and introduce spherical and simplicial realisations of this algorithm. The link between SAS and PLS is made in §4, where we show that for a limited class of functions, PLS is SAS. Numerical results which confirm the theoretical results, and indicate directions for future research, are given in §5.

2 Somewhat Adaptive Search (SAS)

Throughout we consider the global mathematical problem

$$(P) \quad \min_{x \in S} f(x)$$

where f is a real-valued function defined on a convex, compact subset S of R^n . We denote the optimal solution by (x_*, y_*) , where $x_* \in \arg \min_{x \in S} f(x)$ and $y_* = f(x_*)$. It is convenient to define $y^* = \max_{x \in S} f(x)$. We do not require that a unique minimum should exist. If there is more than one, we choose x_* arbitrarily.

We will consider stochastic sequential search procedures whose aim is to locate (x_*, y_*) . The sample path of evaluation points we denote X_1, X_2, \dots , and the associated function values Y_1, Y_2, \dots . Epoch $i > 1$ is said to be a *record* of the sequence

$\{Y_k\}$ if $Y_i < \min\{Y_1, \dots, Y_{i-1}\}$ or $Y_i = y_*$. For technical reasons it is convenient to include the latter condition. Epoch $i = 1$ is always considered to be a record. The corresponding value Y_i is called a *record value*. The k^{th} record value of the sequence of evaluations we denote $Y_{R(k)}$, and the number of iterations from the $(k-1)^{\text{st}}$ to the k^{th} record is denoted by I_k . Thus $I_k = R(k) - R(k-1)$.

We pause to recall the definition of the PAS algorithm for solving problem (P):

Definition 2.1 *Pure Adaptive Search (PAS)*

Initial Step: Set $k = 0$ and $S_0 = S$.

Iterative Step: Increment k

i) *Select evaluation point.*

Choose x_k uniformly distributed on S_{k-1} .

Set $w_k = f(x_k)$.

ii) *Update localisation.*

Set $S_k = f^{-1}(-\infty, w_k) \cup \operatorname{argmin}_{x \in S} f(x)$

Stopping Criterion: Stop if a stopping criterion is met, else return to the iterative step.

Here $f^{-1}(-\infty, w_k) = \{x \in S : -\infty < f(x) < w_k\}$.

Mindful of the virtues of PAS, but aware of the impossibility of achieving it in practice, we now define a new class of algorithms. This is an attempt to keep these virtues while at the same time allowing room to construct practical algorithms. The algorithms require that two conditions should hold. The first allows the algorithm to mark time between records, but not for too long, while the second insists that the quality of the records be as good as those of PAS. The first condition gives the space needed to implement the algorithm, while together they ensure that the “linearity in dimension” drawcard of PAS is retained.

Definition 2.2 *A stochastic sequential search algorithm for solving (P) is termed Somewhat Adaptive Search (SAS) if the following two conditions are satisfied:*

i) *There exists a bound $\beta \geq 1$ such that $E(I_k) \leq \beta$ for all k , and*

ii) *$\{Y_{R(k)} : k = 1, 2, \dots\}$ is stochastically equivalent to $\{W_k : k = 1, 2, \dots\}$, the sequence of records of PAS.*

In [13] Zabinsky and Smith define the *relative improvement* associated with an evaluation y as $z = (y^* - y)/(y - y_*)$. In order to state our main theorem, we extend the language of [13] to:

$N_{SAS}(z)$ = the number of iterations of SAS achieving a relative improvement of z or less
 $N_{SAS}^*(y)$ = the number of iterations of SAS required to achieve a value of y or lower.

The corresponding expressions for PAS we denote by $N_{PAS}(z)$ and $N_{PAS}^*(y)$.

Theorem 2.1 *For SAS applied to problem (P), we have*

$$E[N_{SAS}^*(y)] \leq \beta E[N_{PAS}^*(y)]$$

Corollary 2.1 *Consider all global optimisation problems (P) over a convex feasible region in R^n with diameter at most d , and all functions Lipschitz with parameter at most M . Suppose an algorithm is SAS for this class of problems, and the bound in Definition 2.2 (i) is at most β . Then*

$$E[N_{SAS}^*(y)] \leq \beta + [\beta \ln(Md/(y - y_*))]n$$

That is, the bound is a linear function of the dimension n of the problem.

Proof of Theorem: Since the $Y_{R(k)}$ are stochastically equivalent to the W_k it follows that

$$N_{SAS}(z) = I_1 + I_2 + \dots + I_{N_{PAS}(z)}$$

$$\begin{aligned} \text{Then } E[N_{SAS}(z)] &= E\{E[I_1 + \dots + I_{N_{PAS}(z)} | N_{PAS}(z) \text{ fixed}]\} \\ &\leq E[\beta N_{PAS}(z)] \\ &= \beta E[N_{PAS}(z)] \end{aligned}$$

Converting this into a result about N_{SAS}^* , we see

$$\begin{aligned} E[N_{SAS}^*(y)] &\leq E[N_{SAS}^*(y)] - 1 + \beta \\ &= E\left[N_{SAS}\left(\frac{y^* - y}{y - y_*}\right)\right] + \beta \\ &\leq \beta E\left[N_{PAS}\left(\frac{y^* - y}{y - y_*}\right)\right] + \beta \\ &= \beta E[N_{PAS}^*(y)], \quad \text{as required.} \end{aligned}$$

Proof of Corollary: When the conditions of Corollary 2.1 hold, it was shown in [13, Theorem 5.3] that $E[N_{PAS}^*(y)] \leq 1 + \lceil \ln(Md/(y - y_*)) \rceil n$. Coupled with the result of Theorem 2.1, this gives the statement in the corollary. \square

We pause to discuss a special case of SAS, namely “ ρ -adaptive search”. Informally, a search is ρ -adaptive, for some ρ , with $0 \leq \rho \leq 1$, if at each iteration the probability that it behaves as PAS is ρ . The letter ρ is chosen as a reminder of “record”.

Definition 2.3 *Let $0 \leq \rho \leq 1$. A stochastic sequential search for solving (P) is termed ρ -adaptive if for each iteration k , and for all sample paths x_1, \dots, x_{k-1} ,*

$$P[\text{algorithm acts as PAS at } k^{\text{th}} \text{ iteration} \mid x_1, \dots, x_{k-1}] \geq \rho$$

In this framework, PRS becomes 0-adaptive, and PAS 1-adaptive. This language gives us a way of describing a spectrum of algorithms between these two extremes. We denote $N_{SAS}^*(y)$ by $N_\rho^*(y)$ in the special case of a ρ -adaptive algorithm. Our next result shows that a non-degenerate ρ -adaptive algorithm is always SAS.

Theorem 2.2 *A ρ -adaptive algorithm, with $0 < \rho \leq 1$, is SAS, with $\beta = 1/\rho$.*

Proof: The definition of ρ -adaptivity ensures that after any iteration k , and independent of the sample path, the probability of a record is greater than or equal to ρ . Thus $E(I_k)$ is less than or equal to the mean of a geometric distribution, with parameter ρ . Thus $E(I_k) \leq 1/\rho$, for all k . Condition ii), that the $Y_{R(k)}$ are stochastically equivalent to the W_k , follows via a straightforward modification of [13, Lemma 3.1]. \square

This completes the setting up of the attainable ideal.

3 Pure Localisation Search (PLS)

We turn our attention now to a new and readily implemented algorithm for solving problem (P). In Theorem 3 we show that the records it produces are stochastically equivalent to those of PAS, while in Theorem 5 we show that it realises a SAS algorithm on a particular class of functions of a single variable. In spirit, the algorithm is a probabilistic analogue of the well-known Piyavskii-Shubert algorithm. We present the algorithm in a general setting initially.

The central idea is the following. An exact tracking of the level set of PAS is an impossible task. Tracking a superset of it is not. Certain “removal” algorithms in the literature (for example [7, 9, 5, 12]), while deterministic, do yield a localisation for the level set at each iteration.

Definition 3.1 *Pure Localisation Search (PLS)*

Initial Step: Set $k = 0$ and $L_0 = S$.

Set $\alpha_0 = \infty$.

Iterative Step: Increment k

i) *Select evaluation point.*

Choose x_k uniformly on L_{k-1} .

Set $y_k = f(x_k)$.

ii) *Update localisation.*

$$\text{Set } \alpha_k = \begin{cases} y_k, & \text{if } y_k < \alpha_{k-1} \\ \alpha_{k-1}, & \text{otherwise.} \end{cases}$$

Set $L_k = L_{k-1} - R_k$, where the removal region R_k is such that

$$R_k \subseteq S - f^{-1}(-\infty, \alpha_k).$$

Stopping Criterion: Stop if a stopping criterion is met, else return to the iterative step.

We define $S_y = f^{-1}(-\infty, y)$. Observe that the special case of PLS with $R_k = \emptyset$ is PRS, while PLS becomes PAS when $R_k = S - S_{\alpha_k}$, so $L_k = S_{\alpha_k}$. It follows readily from Theorem 3.2 of this section that PLS converges with probability one. An important observation concerning any PLS is that $L_k \supseteq S_{\alpha_k}$, or in words, the localisation contains the level set.

That PLS always has the second property of SAS is shown in the next theorem.

Theorem 3.1 *For PLS applied to the global optimisation problem (P), the stochastic process of PLS records is equal in distribution to the stochastic process of PAS records. That is*

$$\{Y_{R(k)} : k = 1, 2, \dots\} \sim \{W_k : k = 1, 2, \dots\}$$

Proof: The proof is an extension of [13, Lemma 3.1]. First we show that the conditional distributions are equal. Let k be any iteration and take $y_* \leq y < y' \leq y^*$. Note that

$$P[W_{k+1} < y | W_k = y'] =$$

$$P[Y_{R(k)+j} < y | Y_{R(k)} = y' \text{ and } Y_{R(k)+1} \geq y' \dots Y_{R(k)+j-1} \geq y' \text{ and } Y_{R(k)+j} < y']$$

since both sides equal $\lambda(S_y)/\lambda(S_{y'})$. Here λ denotes Lebesgue measure on R^n . Then we have

$$\begin{aligned} P[Y_{R(k)+1} < y | Y_{R(k)} = y'] &= P[Y_{R(k)+1} < y | Y_{R(k)} = y'] + P[Y_{R(k)+1} \geq y' \text{ and } Y_{R(k)+2} < y | Y_{R(k)} = y'] + \dots \\ &= P[Y_{R(k)+1} < y' | Y_{R(k)} = y'] \cdot P[Y_{R(k)+1} < y | Y_{R(k)} = y' \text{ and } Y_{R(k)+1} < y'] \\ &\quad + P[Y_{R(k)+1} \geq y' | Y_{R(k)} = y'] \cdot P[Y_{R(k)+2} < y' | Y_{R(k)} = y' \text{ and } Y_{R(k)+1} \geq y'] \\ &\quad + P[Y_{R(k)+2} < y | Y_{R(k)} = y' \text{ and } Y_{R(k)+1} \geq y' \text{ and } Y_{R(k)+2} < y'] + \dots \\ &= [p_1 + (1 - p_1)p_2 + (1 - p_1)(1 - p_2)p_3 + \dots] P[W_{k+1} < y | W_k = y'] \end{aligned}$$

where $p_i = P[Y_{R(k)+i} < y' | Y_{R(k)} = y' \text{ and } Y_{R(k)}, \dots, Y_{R(k)+i-1} \geq y']$, the probability that the first record after the k^{th} record occurs at epoch $R(k) + i$. Now the sequence $\{p_i\}$ is bounded away from zero, since $p_i \geq \lambda(S_{y'})/\lambda(S) > 0$ for all i . It follows by an elementary argument that $p_1 + (1 - p_1)p_2 + \dots = 1$. Thus

$$P[Y_{R(k)+1} < y | Y_{R(k)} = y'] = P[W_{k+1} < y | W_k = y']$$

We now use induction to show that the unconditional distributions are equal. By convention, $R(1) = 1$ and from the definition of PLS, $P[Y_1 < y] = P[W_1 < y]$ for all y , for $y_* \leq y \leq y^*$. Hence $Y_{R(1)} \sim W_1$.

Now consider $k > 1$ and suppose that $Y_{R(i)} \sim W_i$ for $i = 1, 2, \dots, k$. Then, for all $y_* \leq y \leq y^*$, we have

$$\begin{aligned} P[Y_{R(k)+1} < y] &= \int_{y_*}^{y^*} P[Y_{R(k)+1} < y | Y_{R(k)} = t] dF_{Y_{R(k)}}(t) \\ &= \int_{y_*}^{y^*} P[W_{k+1} < y | W_k = t] dF_{W_k}(t) \\ &= P[W_{k+1} < y] \end{aligned}$$

The second equality follows using the equality of conditional distributions and the induction hypothesis. By induction it follows that the two sequences are equal in marginal distribution, hence in joint distribution, as required. \square

The way in which PLS is sandwiched between PRS and PAS is made clear in the next theorem.

Theorem 3.2 Fix f as in problem (P), and fix a relative improvement level, $z > 0$. Let $N_{PAS}(z)$, $N_{PLS}(z)$ and $N_{PRS}(z)$ be the number of iterations of PAS, PLS and PRS respectively achieving a relative improvement of z or less. Then

$$P[N_{PAS}(z) < k] \geq P[N_{PLS}(z) < k] \geq P[N_{PRS}(z) < k]$$

Corollary 3.1

- i) $E[N_{PAS}(z)] \leq E[N_{PLS}(z)] \leq E[N_{PRS}(z)]$
- ii) $N_{PAS}^{1-\alpha}(z) \leq N_{PLS}^{1-\alpha}(z) \leq N_{PRS}^{1-\alpha}(z)$ where $N_{alg}^{1-\alpha}(z)$ is the number of iterations of the algorithm required to achieve a relative improvement of z with probability not less than $1 - \alpha$.

Proof of Theorem: Let y correspond to the relative improvement of z . Then

$$\begin{aligned} P[N_{PAS}(z) < k] &= P[W_k \leq y] \\ &= P[Y_{R(k)} \leq y], \text{ by Theorem 3.1} \\ &\geq P[N_{PLS}(z) < k]. \end{aligned}$$

since if PLS achieves a relative improvement of z before the k^{th} iteration $Y_k \leq y$, whence $Y_{R(k)} \leq y$.

In order to show that $P[N_{PLS}(z) < k] \geq P[N_{PRS}(z) < k]$ we now show that $P[N_{PLS}(z) > k - 1] \leq P[N_{PRS}(z) > k - 1]$. Note that $i = N_{PLS}(z) + 1$ is the first epoch such that $x_i \in S_y$. Thus, for any k , $k = 1, 2, \dots$,

$$\begin{aligned} P[N_{PLS}(z) > k - 1] &= P[x_1 \notin S_y \text{ and } x_2 \notin S_y \dots \text{ and } x_k \notin S_y] \\ &= P[x_1 \notin S_y] P[x_2 \notin S_y | x_1 \notin S_y] \dots P[x_k \notin S_y | x_1, \dots, x_{k-1} \notin S_y] \end{aligned}$$

Consider a PLS sample path with first $j-1$ domain points not in S_y , where $j \in \{1, \dots, k\}$. Then $\alpha_{j-1} \geq y$, so $S_{\alpha_{j-1}} \supseteq S_y$, or $L_{j-1} \supseteq S_y$. It follows that

$$P[x_j \in S_y | x_1, \dots, x_{j-1}] = \lambda(S_y) / \lambda(L_{j-1}) \geq \lambda(S_y) / \lambda(S)$$

Hence,

$$P[x_j \in S_y | x_1, \dots, x_{j-1} \notin S_y] \geq \lambda(S_y) / \lambda(S)$$

as it is the average of the above term over all initial segments x_1, \dots, x_{j-1} of PLS sample paths with domain points not in S_y . Thus

$$P[N_{PLS}(z) > k - 1] \leq [1 - \lambda(S_y)/\lambda(S)]^k = P[N_{PRS}(z) > k - 1]$$

Remarks

1. When f is Lipschitz continuous with Lipschitz bound M , two realisations of PLS immediately arise. The first is a stochastic analogue of [5], the second a stochastic analogue of [12]. In the former,

$$R_k = \bigcup_{i=1, \dots, k} B_i$$

where B_i is the ball of radius $(y_i - \alpha_k)/M$, centered at x_i . In the latter,

$$R_k = \bigcup_{i=1, \dots, k} C_i$$

where C_i is a standard simplex of radius $(y_i - \alpha_k)/M$, centered at x_i . We call the former *Spherical* PLS, and the latter *Simplicial* PLS. Note that when $n = 1$ these two realisations of PLS reduce to the same algorithm.

2. The simplicial and spherical realisations of PLS mentioned above are not trivial to implement. When $n = 1$ it is necessary to store a linked list of the intervals which comprise L_k , so that selecting the next evaluation point is performed by choosing a random number in $[0,1]$ and moving through the intervals to the x_k value. Updating the localisation involves an updating of the linked list. For $n > 1$, Spherical PLS has the virtue of producing a tighter localisation than Simplicial PLS, since the removed ball always contains the removed simplex. On the other hand, choosing x_k in Spherical PLS has so far been achieved through an acceptance-rejection approach, whereas with Simplicial PLS a linked list of simplex tops can be stored, and a procedure similar to the $n = 1$ case used to find x_k , see [12, 11].
3. In [12] the term “bracket” was used to describe the $n+1$ dimensional region known to contain the global minima. When $n = 1$, PLS yields a bracket composed of disjoint similar triangles, whereas in Simplicial PLS it is a union of overlapping,

but similar, simplexes. For Spherical PLS the bracket is more complicated to describe. The projection of this bracket onto the domain is what we term the localisation.

4 Linking the ideal of SAS to the reality of PLS

Is it possible to find a class of functions and a natural algorithm which achieves SAS? This section is devoted to showing that the answer is, surprisingly, yes.

We begin by defining a function of a single variable which we call, for obvious reasons, the (upside down) “witch’s hat”. For $h \in [0, 1]$ we define the witch’s hat of height h to be $w_h(x) = \min(|x|, h)$, for $x \in [-1, 1]$. Note that w_h is Lipschitz, with Lipschitz constant $M = 1$. That is, $w_h \in L(1)$, the class of all functions with $M = 1$ on their given domain. It will be convenient to call the graph of w_h on $[-1, -h] \cup [h, 1]$ the “brim”, and w_h on $[-h, h]$ the “cap”. We now define the class of functions, C_h , as all those $f \in L(1)$ which agree with w_h on $[-h, h]$ and elsewhere on $[-1, 1]$ lie above w_h . When PLS is applied to a function in C_h , the localisation eventually becomes the level set. This is formalised in the following theorem.

Theorem 4.1 *Let f be any function in C_h , and for PLS with $M=1$, let N be the number of iterations until the localisation becomes the level set. Then*

$$E[N] < 6 + 26/h$$

Before presenting the proof, we proceed to the consequence:

Corollary 4.1 *For functions in C_h , PLS is SAS, with $\beta = 6 + 26/h$.*

Proof of Corollary: Partition the sample paths as $\cup_{i=1}^{\infty} \Omega_i$, where Ω_i is the set of all sample paths for which the localisation becomes the level set at the i^{th} iteration. Then for $\omega \in \Omega_i$, a fortiori $I_k(\omega) \leq i$ for all k , since for such sample paths there are at most i iterations from any one record to the next. Thus

$$E(I_k) \leq \sum_{i=1}^{\infty} P[\Omega_i] i = E[N]$$

for any k . It follows that condition i) of SAS is satisfied with $\beta = 6+26/h$, using Theorem 4.1. Condition ii) of SAS follows immediately from Theorem 3.1. \square

Proof of Theorem: Take $f \in C_h$. A typical situation which would arise when running PLS on f , once an evaluation is found less than h , is shown in Figure 1.

Figure 1 here

The localisation, L_k , consists of four parts:

1. The level set, A_k .
2. A finite union of intervals, B_k , under the brim.
3. One interval, C_k^l , to the left of A_k and under the cap.
4. One interval, C_k^r , to the right of A_k and under the cap.

These sets are indicated in Figure 1. Denote the total length of these sets by a, b, c^l , and c^r respectively. We say at any stage that the run is *cap separated* if there has been an evaluation under the cap both to the left and to the right of the origin.

Four facts are needed in the final proof of the theorem. We present these now. The first three are readily shown; we give a proof for the fourth.

Fact 1 *If the run at the k^{th} iteration is cap separated, then the depth of the bracket over the complement of A_k is less than or equal to $b + c^l + c^r$.*

Fact 2 *Denote by d the depth of the bracket over the complement of A_k . Then*

$$P[B_{k+1} \cup C_{k+1}^l \cup C_{k+1}^r = \phi \mid x_1, \dots, x_k, \text{ and that } x_{k+1} \in A_k] \geq (a - 2d)/a$$

where a and d are the values after the k^{th} iteration.

Fact 3

$$P[C_{k+1}^l = \phi \mid x_1, \dots, x_k, \text{ and that } x_{k+1} \in C_k^l] \geq 1/2$$

Fact 4 *Consider $t \in (0, h)$. If α_k , the lowest evaluation immediately after the k^{th} iteration, is less than t , then the number of further iterations under the brim is less than or equal to $2 \left\lceil \frac{1-h}{h-t} \right\rceil$, where $\lceil x \rceil$ is the least integer greater than or equal to x .*

Proof of Fact 4: Suppose that after k iterations we have $\alpha_k < t < h$. Let z_1, z_2, \dots be the later iterations of PLS which are in $[h, 1]$. If $i < j$ then $z_j \notin \overline{B}_{h-t}(z_i)$, the closed ball of radius $h - t$ centered at z_i , so $\{\overline{B}_{(h-t)/2}(z_i) : i = 1, 2, \dots\}$ is a mutually disjoint collection of closed intervals whose union is a subset of $[h - (h - t)/2, 1 + (h - t)/2]$. It follows that the collection must be finite, having say m elements, and furthermore, that $m(h - t) < 1 - t$. Thus m is less than or equal to the biggest integer less than $(1 - t)/(h - t) = (1 - h)/(h - t) + 1$. This is $\left\lfloor \frac{1-h}{h-t} \right\rfloor$. Fact 4 then follows by doubling this figure.

The heart of the proof of the theorem rests in recognizing that if we count N_t , the number of iterations until the lowest known evaluation is less than t , and also the number of subsequent iterations, N_p , until we can be sure that the localisation is the level set, then $N_t + N_p$ is greater than or equal to N .

Following the iteration N_t at which $\alpha_{N_t} < t$, we define five types of "progress" event which can occur. These are:

" P_1 " Cap separation occurs for the first time at the $(k + 1)^{th}$ iteration.

" P_2 " $C_k^l \neq \phi$ and $C_{k+1}^l = \phi$

" P_3 " $C_k^r \neq \phi$ and $C_{k+1}^r = \phi$

" P_4 " $x_{k+1} \in B_k$

" P_5 " $B_{k+1} \cup C_{k+1}^l \cup C_{k+1}^r = \phi$

Informally, a progress step is a movement towards the localisation becoming the level set, progress step five. Note that steps one, two and three can occur only once, while step four can occur at most $2\lceil(1 - h)/(h - t)\rceil$ times. Thus, once there has been $2\lceil(1 - h)/(h - t)\rceil + 3$ progress steps following iteration N_t , the localisation must equal the level set. If we let

N_t = the number of iterations, k , until $\alpha_k < t$, and

N_p = the number of iterations following the N_t^{th} iteration
to achieve $2\lceil(1 - h)/(h - t)\rceil + 3$ progress steps,

we have $N \leq N_t + N_p$, so $E[N] \leq E[N_t] + E[N_p]$.

Certainly $E[N_t]$ is smaller for PLS than PRS on f . For PRS on f , the distribution of the number of iterations until a value less than or equal to t is geometric, with probability t . Thus $E[N_t] \leq 1/t$.

We conclude the proof by showing that once we have $\alpha_{N_t} < t$, then the probability of a progress step is always at least $1/6$. The distribution of N_p is negative binomial, so $E[N_p] \leq 6(2 \lceil (1-h)/(h-t) \rceil + 3)$, whence

$$E[N] \leq \frac{1}{t} + 6(2 \lceil \frac{1-h}{h-t} \rceil + 3) = 18 + \frac{1}{t} + 12 \lceil \frac{1-h}{h-t} \rceil$$

Putting $t = h/2$ demonstrates the statement in the theorem.

In order to show that the probability of progress is always greater than or equal to $1/6$, we consider three cases. We suppose we have an initial segment of x_1, \dots, x_k , and $\alpha_k \leq t$.

Case 1: The bracket is not cap separated. Then

$$\begin{aligned} & P[\text{progress at } (k+1)^{\text{st}} \text{ iteration}] \\ & \geq P[P_1] + P[x_{k+1} \in C_k^l \text{ and } P_2] + P[x_{k+1} \in C_k^r \text{ and } P_3] + P[P_4] \\ & \geq \frac{a/2}{a+c^l+c^r+b} + \frac{c^l/2}{a+c^l+c^r+b} + \frac{c^r/2}{a+c^l+c^r+b} + \frac{b}{a+c^l+c^r+b} \\ & \geq \frac{1}{2} \end{aligned}$$

Case 2: The bracket is cap separated, and $\frac{a}{a+c^l+c^r+b} \geq \frac{2}{3}$. Then

$$\begin{aligned} & P[\text{progress at } (k+1)^{\text{st}} \text{ iteration}] \\ & \geq P[x_{k+1} \in A_k \text{ and } P_5] + P[x_{k+1} \in C_k^l \text{ and } P_2] + P[x_{k+1} \in C_k^r \text{ and } P_3] + P[P_4] \\ & \geq \frac{a-2(c^l+c^r+b)}{a+c^l+c^r+b} + \frac{c^l/2}{a+c^l+c^r+b} + \frac{c^r/2}{a+c^l+c^r+b} + \frac{b}{a+c^l+c^r+b} \\ & = \frac{a-3/2(c^l+c^r+b)+b/2}{a+c^l+c^r+b} \\ & \geq \frac{a}{a+c^l+c^r+b} - \frac{3}{2} \frac{c^l+c^r+b}{a+c^l+c^r+b} \\ & \geq \frac{2}{3} - \frac{3}{2} \frac{1}{3} = \frac{1}{6} \end{aligned}$$

Case 3: The bracket is cap separated, and $\frac{a}{a+c^l+c^r+b} < \frac{2}{3}$. Then

$$\begin{aligned} & P[\text{progress at } (k+1)^{\text{st}} \text{ iteration}] \\ & \geq P[x_{k+1} \in C_k^l \text{ and } P_2] + P[x_{k+1} \in C_k^r \text{ and } P_3] + P[P_4] \end{aligned}$$

$$\begin{aligned}
&\geq \frac{c^l/2}{a+c^l+c^r+b} + \frac{c^r/2}{a+c^l+c^r+b} + \frac{b}{a+c^l+c^r+b} \\
&\geq \frac{1}{2} \cdot \frac{c^l+c^r+b}{a+c^l+c^r+b} \\
&\geq \frac{1}{2} \cdot \frac{1}{3} = \frac{1}{6} \quad \square
\end{aligned}$$

5 Numerical results

This section gives empirical support to results in the previous sections. Although the emphasis of this paper is on showing PLS is an effectively implementable stochastic variant of the Piyavskii-Shubert algorithm which is SAS in some situations, we begin with a comparison of the various algorithms with the Piyavskii-Shubert algorithm. The second set of numerical tests show the bounds proved in Theorem 4.1 and Corollary 4.1 are conservative. The last tests show that for the higher dimensional analogue of the witch's hat, Simplicial PLS keeps some of the similarities from the one dimensional case. Evidence, however, suggests that even in this simple case the number of function evaluations to convergence is not linear with dimension.

Comparison of PAS, PLS, PRS and the Piyavskii-Shubert algorithm in dimension one

Generally speaking, compared with PLS and PRS, Piyavskii-Shubert usually takes fewer function evaluations. However for functions with a large number of nearly equal global minima, Simplicial PLS can on average require less work than the Piyavskii-Shubert algorithm to attain modest accuracy. Theorem 3.2 is empirically verified. It is interesting to note that the work required by the Piyavskii-Shubert algorithm and its stochastic variant, PLS, is very similar to that required by the theoretical PAS.

Two random selections of functions were made. We obtain the number of iterations until the global minimum is found to a specified tolerance, using the various algorithms.

The first selection consisted of 69 Lipschitz continuous functions with $M=1$ that usually had a small number of local minima, generally one. These were produced by an obvious modification of a procedure due to Graf, Mauldin and Williams described in [4, p.240-241]. Figure 2 shows that for this class the algorithms ranked from best to worst are PAS, Piyavskii-Shubert, PLS and PRS.

Insert Figure 2 here

The second selection consisted of 50 Lipschitz continuous functions with $M=1$, of the form $(1/A)\sin(Ax+B)$ where A and B were randomly chosen. In this selection all of the functions have between one and eight global minima. Table 1 shows that for this class the algorithms ranked from best to worst are PAS, PLS, Piyavskii-Shubert, and PRS for modest accuracy, but PAS, Piyavskii-Shubert, PLS and PRS when greater accuracy is required.

ϵ	PAS	Piyavskii-Shubert	PLS	PRS
0.1/A	3.0	5.6	5.5	7.6
0.01/A	4.2	9.6	11.2	23.9

Table 1: Mean number of evaluations to convergence, with the two relative accuracy levels, for the 50 sinusoidal functions

Observed and theoretical results for PLS on witch's hats in dimension one

Theorem 4.1 gave a theoretical upper bound on the average number of iterations until the localization becomes the level set for the witch's hat. Table 2 compares this with the average observed number of iterations, over 1000 runs, thus showing that the theoretical bound is roughly ten times too large. The observed values of $E[N]$ are conservative estimates for β , as demonstrated in Corollary 4.1. Empirical tests have shown that β is roughly one third of $E[N]$.

h	Observed	Theoretical
1	4.8	32
1/2	7.4	58
1/3	9.8	84
1/4	12.1	110
1/8	21.4	214

Table 2: A comparison of the observed and the theoretical average number of iterations until the localisation becomes the level set, for the witch's hat with varying values of h

Observed results for Simplicial PLS on witch’s hat in higher dimensions

One higher dimensional analogue of the witch’s hat is the upward facing simplicial cone defined over a simplicial domain. For this function with Simplicial PLS, the localization, as in dimension one, becomes the level set.

Dimension	Average observed no. iterations	
	Localisation = level set	Tolerance of 0.1
1	4.8	4.9
2	15	9.6
3	37	17.4
4	–	29.0

Table 3: The behaviour of Simplicial PLS on higher dimensional analogues of the witch’s hat

The second column of Table 3 shows the average observed number of iterations, for 100 trials, for this to happen when the dimension is 1, 2 and 3. The third column of Table 3 shows the expected number of iterations for convergence, to within $\epsilon = 0.1$ of the global minimum, for Simplicial PLS for dimensions 1, 2, 3 and 4. Evidently these numbers are not linear in dimension. For PAS it is shown in [13] that this quantity is linear in dimension, so this implementation of PLS is not “uniformly” close enough to PAS to maintain linearity in dimension. Empirically it appears that PLS is SAS for the witch’s hat in dimensions greater than one, but evidence suggests that the β values are unbounded. If there were a bound, then PLS would be linear in dimension (Corollary 2.1). However, if the β values prove to be bounded by a function that is polynomial in dimension, then polynomial complexity of PLS would result.

Acknowledgment: For encouragement and helpful discussions, the authors would like to thank Murray Smith, Stephen Samuels and Roger Wets.

References

- [1] W. Baritompä, “Customizing methods for global optimization — a geometric viewpoint”, *J. Global Optimization* (1992) (to appear).

- [2] P. Basso, "Iterative methods for the localization of the global maximum", *SIAM J. Numer. Anal.* 19 (1982) 781-792.
- [3] C.J.P. Be'lisle, H.E. Romeijn and R.L. Smith, "Hide-and-Seek: A Simulated Annealing Algorithm for Global Optimization", Working paper, Department of Industrial and Operations Engineering, The University of Michigan (1990).
- [4] S. Graf, R. Daniel Mauldin and S.C. Williams, "Random homeomorphisms", *Advances in Mathematics* 60 (1986) 239-359.
- [5] R.H. Mladineo, "An algorithm for finding the global maximum of a multimodal, multivariate function", *Mathematical Programming* 34 (1986) 188-200.
- [6] N. Patel, R.L. Smith, and Z.B. Zabinsky, "Pure Adaptive Search in Monte Carlo Optimization", *Mathematical Programming* 43 (1988) 317-328.
- [7] S.A. Piyavskii, "An algorithm for finding the absolute extremum of a function", *USSR Comp. Math. and Math. Phys.* 12 (1972) 57-67.
- [8] A.H.G. Rinnooy Kan and G.T. Timmer, "Global Optimization", Technical Report 8612/A, Econometric Institute, Erasmus University, Rotterdam (1986).
- [9] B.O. Shubert, "A sequential method seeking the global maximum of a function", *SIAM J. Numer. Anal.* 9 (1972) 379-388.
- [10] F.J. Solis and R.J-B. Wets, "Minimization by random search techniques", *Mathematics of Operations Research* 6 (1981) 19-30.
- [11] G.R. Wood, "Multidimensional bisection applied to global optimisation", *Computers and Mathematics with Applications* 21 (1991) 161-172.
- [12] G.R. Wood, "The bisection method in higher dimensions", *Mathematical Programming* (1992) (to appear).
- [13] Z.B. Zabinsky and R.L. Smith, "Pure Adaptive Search in global optimization", *Mathematical Programming* 53 (1992) 323-338.
- [14] Z.B. Zabinsky, R.L. Smith, J.F. McDonald, H.E. Romeijn and D.E. Kaufman "Improving Hit-and-Run for global optimization", *J. Global Optimization* (1992) (to appear).

(WPB, GRW and ZBP)

Department of Mathematics,
University of Canterbury
Christchurch, New Zealand.

(RHM)

Rider College,
Lawrenceville, New Jersey, USA.

(ZBZ)

Industrial Engineering Program,
University of Washington,
Seattle, Washington 98195, USA.

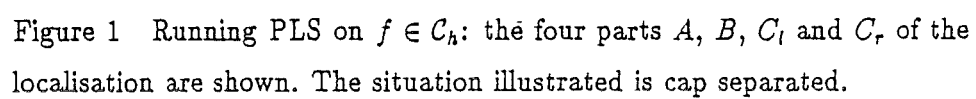


Figure 1 Running PLS on $f \in \mathcal{C}_h$: the four parts A , B , C_l and C_r of the localisation are shown. The situation illustrated is cap separated.

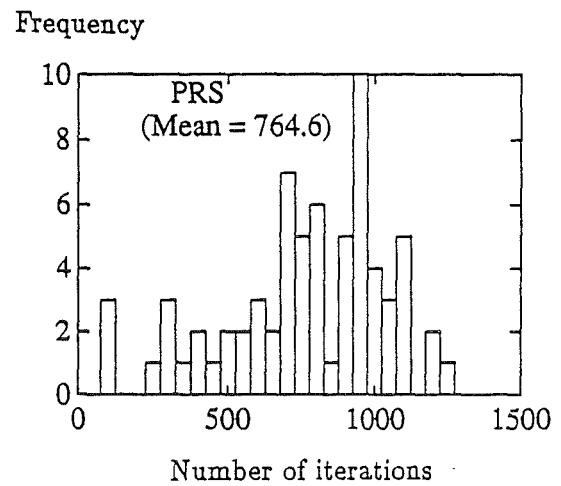
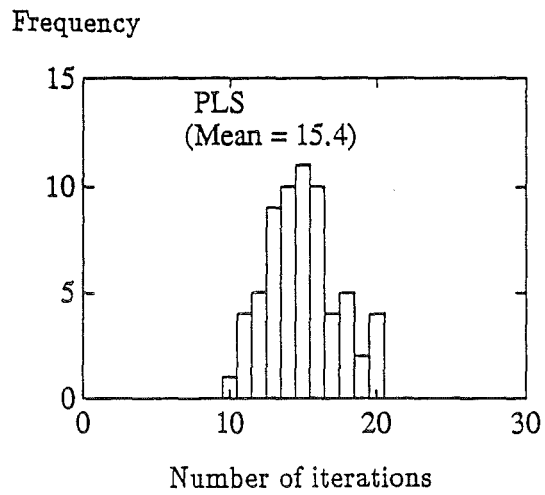
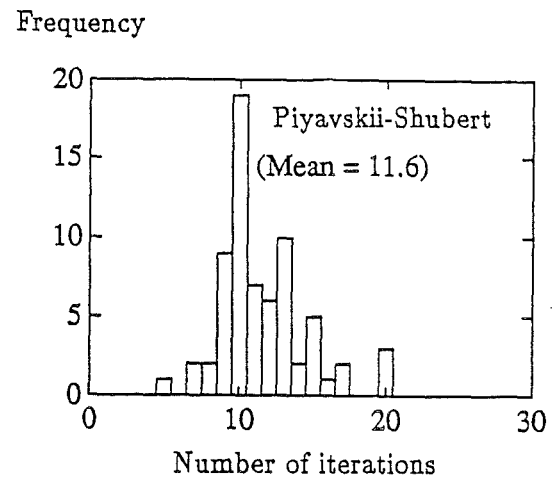
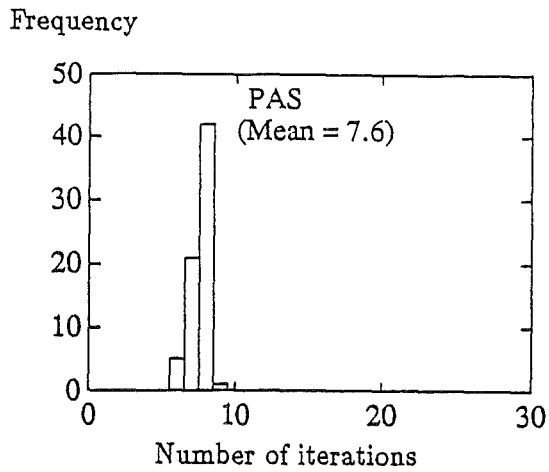


Figure 2 Histograms showing the mean number of iterations to convergence, with $\epsilon = 0.0005$, for the 69 Graf-generated functions and the four algorithms. Note the different horizontal scale for PRS.